BASE DE DATOS II

JAIME ZAMBRANA CHACÓN

Contenido

TEMA 1: HISTORIA DE LAS BASES DE DATOS 1

DISPOSITIVOS MECANICOS. Cronología 1

3000 A.C. Abaco 1

1612. Varillas de John Napier 2

1622. Reglas de cálculo (William Oughtred) 2

1642. Máquina de Pascal: Pascalina 3

1670. Máquina de Leibnitz 3

1960Tarjetas perforadas 4

1972. Primera calculadora producida en masa 4

Historia de las Bases de datos 5

Década de 1960 5

Década de 1970 6

Década de 1980 7

Década años 1990 7

Siglo XXI 8

TEMA 2: SISTEMA DE GESTIÓN DE BASE DE DATOS 9

Sistema de Gestor de Base de Datos SGBD 10

Objetivos y servicios que brinda un SGBD. 13

Flexibilidad e independencia 13

Problemas de la redundancia 13

Integridad de los datos 13

Concurrencia de usuarios 14

Seguridad 14

TEMA 3: MODELO DE BASES DE DATOS 15
TEMA 4: MODELO ENTIDAD RELACIÓN 18
Entidad y conjunto de entidades 18
Entidad 18
Atributo 19
Relación 20
Restricciones 21
Relación de uno a muchos 21
Relación de muchos a uno 21
Relación de muchos a muchos 22
Resumen 22
TEMA 5: MODELO RELACIONAL 24
Llave primaria 25
Llaves candidatas 25
Llaves foráneas 25
Relaciones 26
Relación de uno a uno 26
Relación de uno a muchos 26
Relación de muchos a muchos 27
TEMA 6: NORMALIZACIÓN DE BASE DE DATOS 28
Introducción 28
Definición de normalización 28
Formas normales ..... **¡Error! Marcador no definido.**
Ejercicios ..... **¡Error! Marcador no definido.**
Ejercicios propuestos ..... **¡Error! Marcador no definido.**

Conclusiones.....**¡Error! Marcador no definido.**

BIBLIOGRAFÍA.....36

Anexo A: ACTIVIDADES.....38

Implementa base de datos que gestionen información para satisfacer requerimientos de negocios, aplicando técnicas de diseño y herramientas actuales considerando metodologías actuales, estándares de la industria y buenas prácticas para su implementación.

TEMA 1: HISTORIA DE LAS BASES DE DATOS

Antes de iniciar este repaso histórico del base de datos (BD), quisiera enfatizar en la importancia de almacenar datos y al mismo comprender funcionamiento de las BD. En la actualidad es común llamar a los datos el **nuevo petróleo**. La analogía no es perfecta. Los datos se pueden usar muchas veces, el petróleo solo una vez. (Harford, 2020)

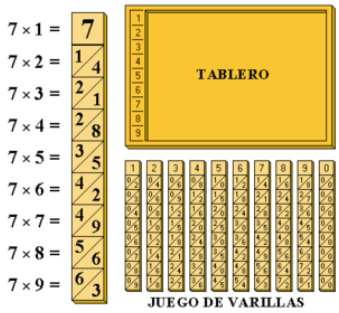
Para entender este fascinante mundo de las BD, se debe conocer sus orígenes, que se remontan a la antigüedad donde ya existían bibliotecas y toda clase de registros que se utilizaban para recoger información sobre las cosechas y principalmente censos, pero su búsqueda era lenta y poco eficaz, esto porque se lo realizaba manualmente, aunque si existía el apoyo de máquinas mecánicas.

DISPOSITIVOS MECANICOS. Cronología

3000 A.C. Abaco

El ábaco es un instrumento de cálculo que sirve para efectuar operaciones aritméticas sencillas (sumas, restas, divisiones y multiplicaciones) y otras más complejas (como calcular raíces).

1612. Varillas de John Napier



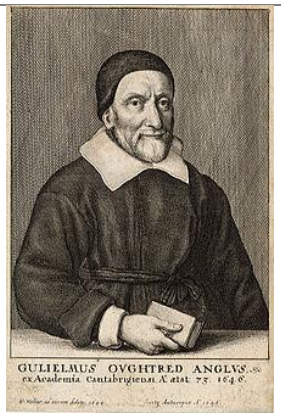
VARILLAS DE NEPER

VER VIDEO

John Napier de Merchiston, llamado también Johannes Neper o Nepair, fue un matemático e inventor escocés, reconocido por ser el primero en definir los logaritmos. También hizo común el uso de la coma decimal en las operaciones aritméticas.

Los huesos de Napier, también conocidos como varillas o bastones de Napier, es un ábaco para realizar operaciones matemáticas simples.

1622. Reglas de cálculo (William Oughtred)



WILLIAM OUGHTRED

VER VIDEO

Antes de que tuviéramos calculadoras, teníamos reglas de cálculo. Las reglas de cálculo fueron inventadas por un ministro episcopal y matemático de nombre William Oughtred.

William Oughtred hizo la primera regla de cálculo al inscribir logaritmos en madera o marfil. Antes de la invención de la calculadora de bolsillo o de mano, la regla de cálculo era una herramienta popular para los cálculos. El uso de reglas de cálculo continuó hasta aproximadamente 1974, después de lo cual las calculadoras electrónicas se hicieron más populares.

1642. Máquina de Pascal: Pascalina

BLAISE PASCAL

VER VIDEO

En 1642, en un esfuerzo por aliviar los interminables y agotadores cálculos y recálculos de su padre de los impuestos adeudados y pagados (en cuyo trabajo se había reclutado al joven Pascal), Pascal, que aún no tenía 19 años, construyó una calculadora mecánica capaz de sumar y restar, llamada calculadora de Pascal o la Pascalina. De las ocho Pascalinas que se sabe que han sobrevivido, cuatro están en poder del Musée des Arts et Métiers en París y una más en el museo Zwinger en Dresde, Alemania, exhiben dos de sus calculadoras mecánicas originales. Aunque estas máquinas son pioneras en otros 400 años de desarrollo de métodos mecánicos de cálculo, y en cierto sentido en el campo posterior de la ingeniería informática, la calculadora no logró ser un gran éxito comercial.

1670. Máquina de Leibnitz

GOTTFRIED LEIBNIZ

VER VIDEO

Leibniz extendió las ideas de Blaise Pascal y, en 1671, introdujo el Staffelwalze (Step Reckoner, también conocido como el Stepped Reckoner o máquina de Leibniz), un dispositivo que, así como ejecutaba adiciones y sustracciones, podía multiplicar, dividir y sacar raíces cuadradas mediante una serie de pasos de adiciones. Los dispositivos de Pascal y Leibniz fueron los antepasados de las computadoras de escritorio de hoy, y los derivados de estas máquinas, incluyendo la calculadora Curta, continuaron siendo producidos hasta que a principios de los años 1970 sus equivalentes electrónicos finalmente llegaron a ser fácilmente disponibles y baratos.

1960Tarjetas perforadas

TARJETAS PERFORADAS

VER VIDEO

La tarjeta perforada o simplemente tarjeta es una lámina hecha de cartulina que contiene información en forma de perforaciones según un código binario. Estos fueron los primeros medios utilizados para ingresar información e instrucciones a una computadora en los años 1960 y 1970. Las tarjetas perforadas fueron usadas con anterioridad por Joseph Marie Jacquard en los telares de su invención, de donde pasó a las primeras computadoras electrónicas. Con la misma lógica se utilizaron las cintas perforadas.

1972. Primera calculadora producida en masa

HP - 35

VER VIDEO

La HP 35 fue la primera calculadora de bolsillo fabricada por la empresa estadounidense Hewlett-Packard y la primera calculadora científica de bolsillo del mundo con funciones trigonométricas, logarítmicas y exponenciales. Como algunas de las calculadoras de escritorio de HP previas, usaba la notación polaca inversa (RPN). Fue presentada al público con un precio de US$395,1 y estuvo disponible desde el 4 de enero de 1972 hasta 1975. Durante ese tiempo, se vendieron más de 300.000 unidades.

Historia de las Bases de datos

Década de 1960

Posteriormente en la época de los sesenta, las computadoras bajaron los precios para que las compañías privadas las pudiesen adquirir; dando paso a que se popularizara el uso de los discos, cosa que fue un adelanto muy efectivo en la época, debido a que a partir de este soporte se podía consultar la información directamente, sin tener que saber la ubicación exacta de los datos.

En esta misma época se dio inicio a las primeras generaciones de bases de datos de red y las bases de datos jerárquicas, ya que era posible guardar estructuras de datos en listas y árboles.

Otro de los principales logros de los años sesenta fue la alianza de IBM y American Airlines para desarrollar SABRE, un sistema operativo que manejaba las reservas de vuelos, transacciones e informaciones sobre los pasajeros de la compañía American Airlines.

Y, posteriormente, en esta misma década, se llevó a cabo el desarrollo del IDS desarrollado por Charles Bachman (que formaba parte de la CODASYL) supuso la creación de un nuevo tipo de sistema de bases de datos conocido como modelo en red que permitió la creación de un standard en los sistemas de bases de datos gracias a la creación de nuevos lenguajes de sistemas de información.

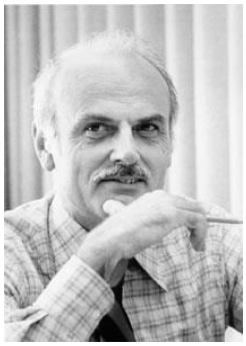
CODASYL (Conference on Data Systems Languages) era un consorcio de industrias informáticas que tenían como objetivo la regularización de un lenguaje de programación estándar que pudiera ser utilizado en multitud de ordenadores.

Los miembros de este consorcio pertenecían a industrias e instituciones gubernamentales relacionadas con el proceso de datos, cuya principal meta era promover un análisis, diseño e implementación de los sistemas de datos más

efectivos; y aunque trabajaron en varios lenguajes de programación como COBOL, nunca llegaron a establecer un estándar fijo, proceso que se llevó a cabo por ANSI.

Década de 1970

Por lo que respecta a la década de los setenta, Edgar Frank Codd, científico informático ingles conocido por sus aportaciones a la teoría de bases de datos relacionales, definió el modelo relacional a la par que publicó una serie de reglas para los sistemas de datos relacionales a través de su artículo “Un modelo relacional de datos para grandes bancos de datos compartidos”.



Este hecho dio paso al nacimiento de la segunda generación de los Sistemas Gestores de Bases de Datos.

Como consecuencia de esto, durante la década de 1970, Lawrence J. Ellison, más conocido como Larry Ellison, a partir del trabajo de Edgar F. Codd sobre los sistemas de bases de datos relacionales, desarrolló el Relational Software System, o lo que es lo mismo, lo que actualmente se conoce como Oracle Corporation, desarrollando así un sistema de gestión de bases de datos relacional con el mismo nombre que dicha compañía.

Posteriormente en la época de los ochenta también se desarrollará el SQL (Structured Query Language) o lo que es lo mismo un lenguaje de consultas o lenguaje declarativo de acceso a bases de datos relacionales que permite efectuar consultas con el fin de recuperar información de interés de una base de datos y hacer cambios sobre la base de datos de forma sencilla; además de analiza grandes cantidades de información y permitir especificar diversos tipos de operaciones frente a la misma información, a diferencia de las bases de datos de los años ochenta que se diseñaron para aplicaciones de procesamiento de transacciones.

Pero cabe destacar que ORACLE es considerado como uno de los sistemas de bases de datos más completos que existen en el mundo, y aunque su dominio

en el mercado de servidores empresariales ha sido casi total hasta hace relativamente poco, actualmente sufre la competencia del SQL Server de la compañía Microsoft y de la oferta de otros Sistemas Administradores de Bases de Datos Relacionales con licencia libre como es el caso de PostgreSQL, MySQL o Firebird que aparecerían posteriormente en la década de 1990.

Década de 1980

Por su parte, a principios de los años ochenta comenzó el auge de la comercialización de los sistemas relacionales, y SQL comenzó a ser el estándar de la industria, ya que las bases de datos relacionales con su sistema de tablas (compuesta por filas y columnas) pudieron competir con las bases jerárquicas y de red, como consecuencia de que su nivel de programación era sencillo y su nivel de programación era relativamente bajo.

Década años 1990

En la década de 1990 la investigación en bases de datos giró en torno a las bases de datos orientadas a objetos. Las cuales han tenido bastante éxito a la hora de gestionar datos complejos en los campos donde las bases de datos relacionales no han podido desarrollarse de forma eficiente. Así se desarrollaron herramientas como Excel y Access del paquete de Microsoft Office que marcan el inicio de las bases de datos orientadas a objetos.

Así se creó la tercera generación de sistemas gestores de bases de datos.

Fue también en esta época cuando se empezó a modificar la primera publicación hecha por ANSI del lenguaje SQL y se empezó a agregar nuevas expresiones regulares, consultas recursivas, triggers y algunas características orientadas a objetos, que posteriormente en el siglo XXI volverá a sufrir modificaciones introduciendo características de XML, cambios en sus funciones, estandarización del objeto sequence y de las columnas autonuméricas. Y además, se creará la posibilidad de que SQL se pueda utilizar conjuntamente con XML, y se

definirá las maneras de cómo importar y guardar datos XML en una base de datos SQL. Dando asi, la posibilidad de proporcionar facilidades que permiten a las aplicaciones integrar el uso de XQuery (lenguaje de consulta XML) para acceso concurrente a datos ordinarios SQL y documentos XML. Y posteriormente, se dará la posibilidad de usar la cláusula order by.

Aunque el boom de la década de los noventa será es el nacimiento del World Wide Web a finales de la década, ya que a través de este se facilitará la consulta a bases de datos.

Siglo XXI

En la actualidad, las tres grandes compañías que dominan el mercado de las bases de datos son IBM, Microsoft y Oracle. Por su parte, en el campo de internet, la compañía que genera gran cantidad de información es Google. Aunque existe una gran variedad de software que permiten crear y manejar bases de datos con gran facilidad, como por ejemplo LINQ, que es un proyecto de Microsoft que agrega consultas nativas semejantes a las de SQL a los lenguajes de la plataforma .NET. El objetivo de este proyecto es permitir que todo el código hecho en Visual Studio sea también orientado a objetos; ya que antes de LINQ la manipulación de datos externos tenía un concepto más estructurado que orientado a objetos; y es por eso que trata de facilitar y estandarizar el acceso a dichos objetos.

Cabe destacar que Visual Studio es un entorno de desarrollo integrado para sistemas operativos Windows que soporta varios lenguajes de programación tales como Visual C++, Visual#, Visual J#, ASP.NET y Visual Basic.NET, aunque se están desarrollando las extensiones necesarias para otros, cuyo objetivo es permitir crear aplicaciones, sitios y aplicaciones web, así como servicios web a cualquier entorno que soporte la plataforma .Net, creando así aplicaciones que intercomuniquen entre estaciones de trabajo, páginas web y dispositivos móviles.

TEMA 2: SISTEMA DE GESTIÓN DE BASE DE DATOS

La mayoría de los sistemas informáticos por no decir todos, utilizan una base de datos para gestionar su información, por lo que es de suma importancia que los desarrolladores de software estén capacitados en el diseño y construcción de bases de datos; en ese sentido es importante partir por definir que es una Base de Datos.

Muchos autores mencionan que la "…colección de datos, normalmente denominada base de datos, contiene información relevante para una empresa"(Silberschatz et al., 2002)

Para Marqués "una base de datos es un conjunto de datos almacenados en memoria externa que están organizados mediante una estructura de datos." (Marqués, 2009).

Por otro lado, Fuentes Gómez (Fuentes Gómez, 2013) define como: "colección de datos, donde los datos están lógicamente relacionados entre sí, tienen una definición y descripción comunes y están estructurados de una forma particular. Una base de datos es también un modelo del mundo real y, como tal, debe poder servir para toda una gama de usos y aplicaciones. [Conference des Statisticiens Européens, 1977]" (Fuentes Gómez, 2013)

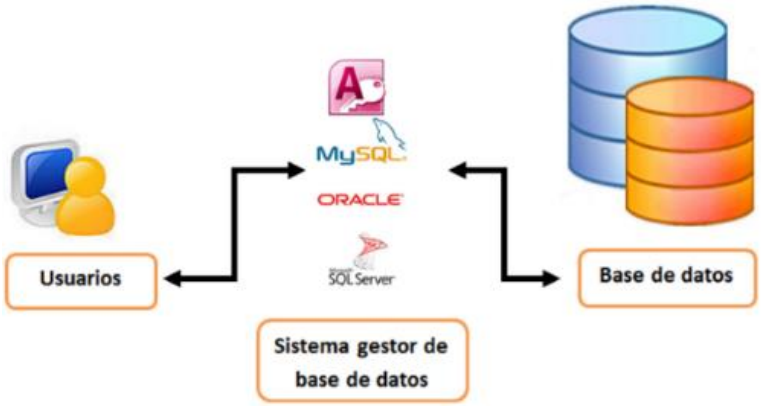
Los autores mencionados, usan términos conceptuales comunes, como **(colección o conjunto de datos)**; por ello es posible considerar a la propia base de datos como una especie de armario electrónico correctamente organizados.

Sistema de Gestor de Base de Datos SGBD

En 1960 no era posible contar con un software que controle que varios usuarios actualizaran datos simultáneamente en una BD; además cabe destacar que la utilización de estos conjuntos de ficheros por parte de los programas de aplicación era excesivamente compleja, de modo que recién en la segunda mitad de los años setenta, fue saliendo al mercado software más sofisticado: los Data Base Management Systems, que a partir de ahora se denominará **sistemas de gestión de Base de Datos (SGBD)**.

Pero, para entender mejor a los SGBD, se debe revisar su evolución histórica desde 1960 hasta nuestros días.

Entre los SGBD de los años sesenta y setenta más representativos están: IMS de IBM, IDS de Bull, DMS de Univac, etc…, estos sistemas trabajaban exclusivamente por lotes (batch), eran sistemas totalmente centralizados, como corresponde a los sistemas operativos de aquellos años; existía un gran ordenador para toda la empresa y una red de terminales tontos.



Para escribir los programas de aplicación se utilizaban lenguajes de alto nivel como Cobol, se disponía también de instrucciones y de subrutinas

especializadas para tratar las BD que requerían que el programador conociese muchos detalles del diseño físico, y que hacían que la programación fuese muy compleja.

Puesto que los programas estaban relacionados con el nivel físico, se debían modificar continuamente cuando se hacían cambios en el diseño y la organización de la BD. La preocupación básica era maximizar el rendimiento: el tiempo de respuesta y las transacciones por segundo.

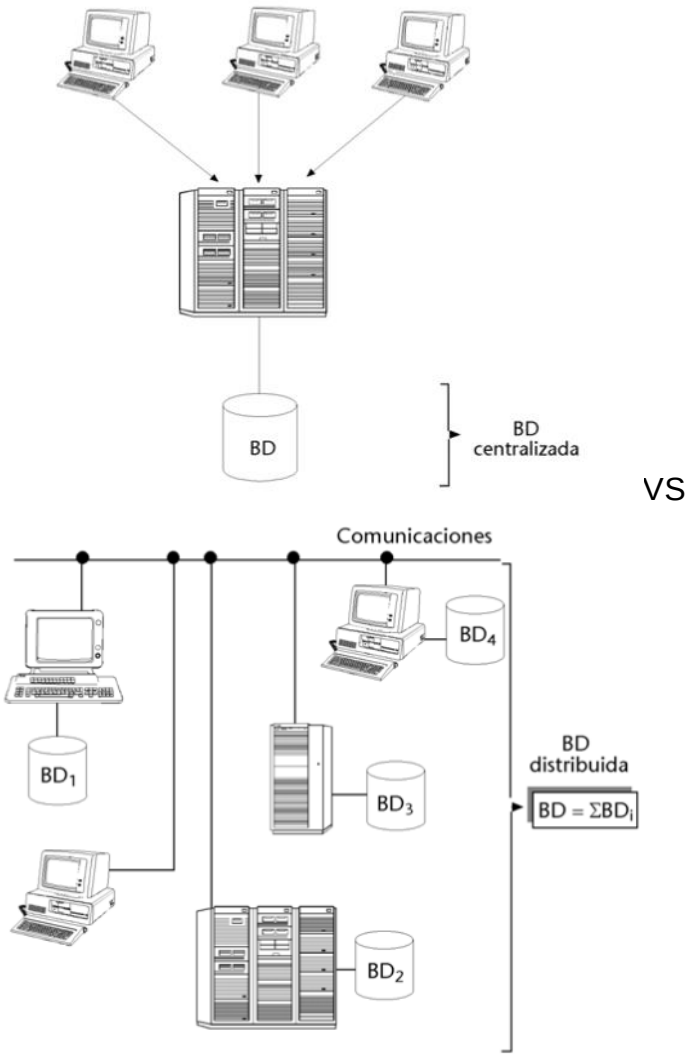
En 1980, los costos de adquisición de un computador personal redujeron, que extendieron la informática a prácticamente todas las empresas e instituciones, esto exigía que el desarrollo de aplicaciones fuese más sencillo. Los SGBD de los años setenta eran demasiado complejos e inflexibles, y sólo los podía utilizar un personal muy cualificado.

La aparición de los SGBD relacionales (nacimiento de Oracle en 1980) supone un avance importante para facilitar la programación de aplicaciones con BD y para conseguir que los programas sean independientes de los aspectos físicos de la BD.

Todos estos factores hacen que se extienda el uso de los SGBD. La estandarización, en el año 1986, del lenguaje SQL produjo una auténtica explosión de los **SGBD relacionales** que prácticamente se usaban en todas las empresas.

En 1990 las empresas ya cuentan con numerosas BD y varios SGBD de diferentes tipos o proveedores, la necesidad de tener una visión global de la empresa y de interrelacionar diferentes aplicaciones que utilizan BD, junto con la facilidad que dan las redes para la intercomunicación entre ordenadores, condujo a un auge de los SGBD, que permitieron que un programa pueda trabajar con diferentes BD como si se tratase de una sola; es lo que se conoce como base de datos distribuida.

Esta distribución ideal se consigue cuando las diferentes BD son soportados por una misma marca de SGBD, es decir, cuando hay homogeneidad, sin embargo, esto no es tan sencillo si los SGBD son heterogéneos; gracias a la estandarización del lenguaje SQL, los SGBD de marcas diferentes pueden darse servicio unos a otros y colaborar para dar servicio a un programa de aplicación. No obstante, en general, en los casos de heterogeneidad no se llega a poder dar en el programa que los utiliza la apariencia de que se trata de una única BD.



Base de datos centralizada vs BD distribuida

Fuente: (Camps Paré et al., 2005)

Objetivos y servicios que brinda un SGBD.

Los SGBD que actualmente están en el mercado pretenden satisfacer un conjunto de objetivos como los siguientes:

Flexibilidad e independencia

La complejidad de las BD y la necesidad de irlas adaptando a la evolución del Sistema de Información hacen que un objetivo básico de los SGBD sea dar flexibilidad a los cambios.

Interesa obtener la máxima independencia posible entre los datos y los procesos usuarios para que se pueda llevar a cabo todo tipo de cambios tecnológicos y variaciones en la descripción de la BD, sin que se deban modificar los programas de aplicación ya escritos ni cambiar la forma de escribir las consultas (o actualizaciones) directas.

Problemas de la redundancia

Si se tiene algo apuntado en dos lugares diferentes no pasará demasiado tiempo hasta que las dos anotaciones dejen de ser coherentes, porque se habrá modificado la anotación en uno de los lugares y se olvida de hacer la anotación en el otro lugar. Un SGBD debiera solucionar este problema.

Integridad de los datos

Nos interesará que los SGBD aseguren el mantenimiento de la calidad de los datos en cualquier circunstancia especialmente cuando el SGBD detecte que un programa quiere hacer una operación que va contra las reglas establecidas al definir la BD, no se lo deberá permitir y le tendrá que devolver un estado de error.

Al diseñar una BD para un SI concreto y escribir su esquema, no sólo definiremos los datos, sino también las reglas de integridad que queremos que el SGBD haga cumplir.

Concurrencia de usuarios

Un objetivo fundamental de los SGBD es permitir que varios usuarios puedan acceder concurrentemente a la misma Base de Datos BD. Cuando un usuario o más de uno están actualizando los datos, se pueden producir problemas de interferencia que tengan como consecuencia la obtención de datos erróneos y la pérdida de integridad de la BD.

Para tratar los accesos concurrentes, los SGBD utilizan el concepto de transacción de BD, concepto de especial utilidad para todo aquello que hace referencia a la integridad de los datos.

Los SGBD deben conseguir que el conjunto de operaciones de una transacción nunca se ejecute parcialmente. O se ejecutan todas, o no se ejecuta ninguna.

Seguridad

El término seguridad se ha utilizado en diferentes sentidos a lo largo de la historia de la informática. Actualmente, en el campo de los SGBD, el término seguridad se suele utilizar para hacer referencia a los temas relativos a la confidencialidad, las autorizaciones, los derechos de acceso, etc.

Estas cuestiones siempre han sido importantes en los SI militares, las agencias de información gubernamentales, etc.

TEMA 3: MODELO DE BASES DE DATOS

Un modelo de base de datos muestra la estructura lógica de la base, incluidas las relaciones y limitaciones que determinan cómo se almacenan los datos y cómo se accede a ellos. Los modelos de bases de datos individuales se diseñan en base a las reglas y los conceptos de cualquier modelo de datos más amplio que los diseñadores adopten. La mayoría de los modelos de datos se pueden representar por medio de un esquema.

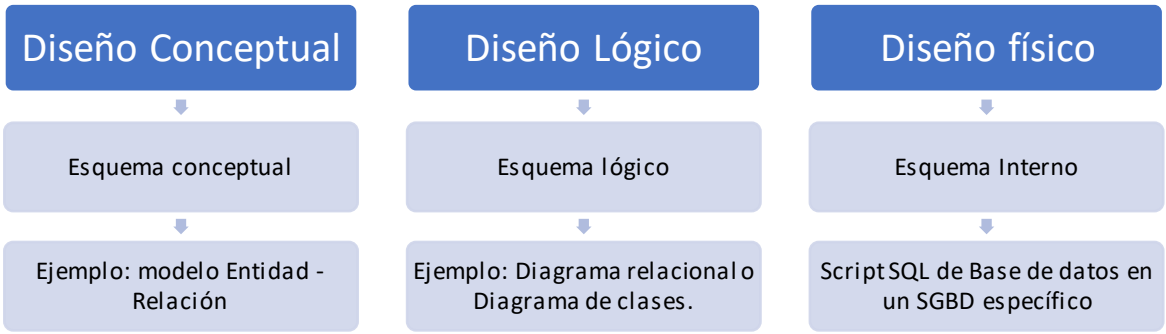
La calidad del análisis y diseño de un sistema de información que pretendemos mecanizar dependerá de los modelos de datos que utilicemos para cada una de las fases de desarrollo. Además, disponer de herramientas software basadas en modelos de datos adecuados a nuestra tarea nos hará más fácil y rentable el diseño y el mantenimiento. Podemos decir que, en líneas generales, el diseño de un sistema de información, en lo que atañe a las bases de datos, (Gómez Ballester et al., 2010, p. 15) tiene tres fases:

Diseño conceptual: en la que se formalizan las estructuras que se observan en el mundo real produciendo lo que se denomina Esquema Conceptual.

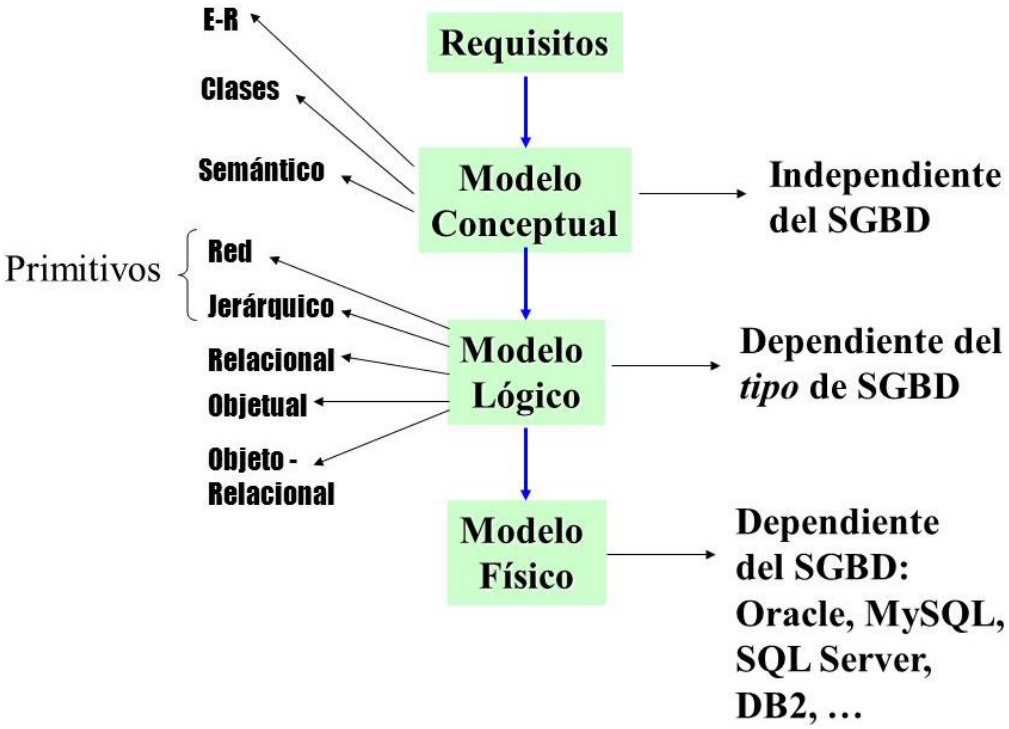
Diseño Lógico: en la que se estructura el conjunto de información de la fase anterior teniendo en cuenta el SGBD que se vaya a utilizar. En esta fase obtendremos el Esquema Lógico.

Diseño Físico: en la que se estructuran los datos en términos de almacenamiento en los dispositivos del ordenador. Es lo que se conoce como Esquema Interno.

Por lo tanto, las fases desde el diseño conceptual al diseño físico son:



Fuente: Creación propia



Los modelos que se estudiaran en este libro son el modelado Entidad Relación **E-R** y el **relacional.**

- Modelo jerárquico
- Modelo en red
- Modelo relacional
- Modelo multidimensional
- Modelo orientada a objetos

- **Modelo Entidad relación**
- **Modelo Relacional**

TEMA 4: MODELO ENTIDAD RELACIÓN

EL modelo de datos entidad-relación al cual se denominará **(E-R)** está basado en una percepción del mundo real consistente en objetos básicos llamados entidades y de relaciones entre estos objetos. Se desarrolló para facilitar el diseño de bases de datos permitiendo la especificación de un esquema de empresas que representa la estructura conceptual completa de una base de datos.

El modelo **E-R** es extremadamente útil para hacer corresponder los significados e interacciones de las empresas del mundo real con un esquema conceptual.

Hay tres nociones básicas que emplea el modelo de datos **E-R:** entidades, conjuntos de relaciones y atributos.

Entidad y conjunto de entidades

Un conjunto de entidades es un conjunto de entidades que comparten las mismas propiedades. Ejemplos: conjuntos de empleados, compañías, clientes, autos, etc.

Ahora bien, se ejemplificará las entidades, atributos, relaciones y por supuesto las líneas que permiten relacionar.

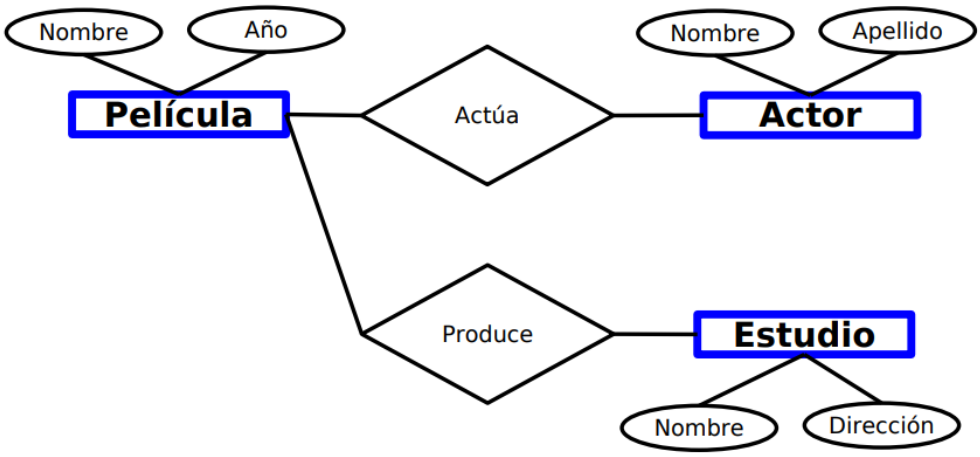
Entidad

Las entidades representan cosas u objetos (ya sean reales o abstractos)

Estas entidades se representan en un diagrama con un rectángulo, como los siguientes.

Una entidad es un elemento con existencia independiente de otros elementos. De manera física puede ser por ejemplo: un auto, una casa, una persona, un empleado., de manera conceptual: un trabajo, un curso, un préstamo.

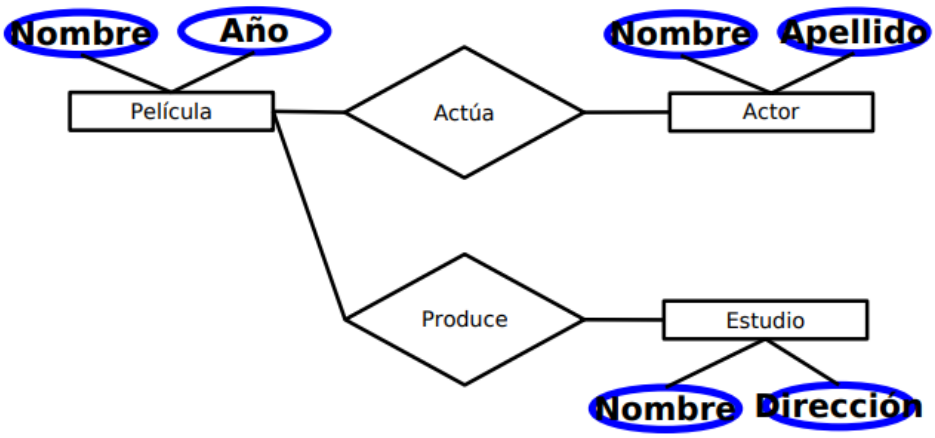
En este ejemplo, Película, Actor y Estudio son entidades:



Atributo

Los atributos describen propiedades que posee cada miembro de un conjunto de entidades. La designación de un atributo para un conjunto de entidades expresa que la base de datos almacena información similar concerniente a cada entidad del conjunto de entidades.

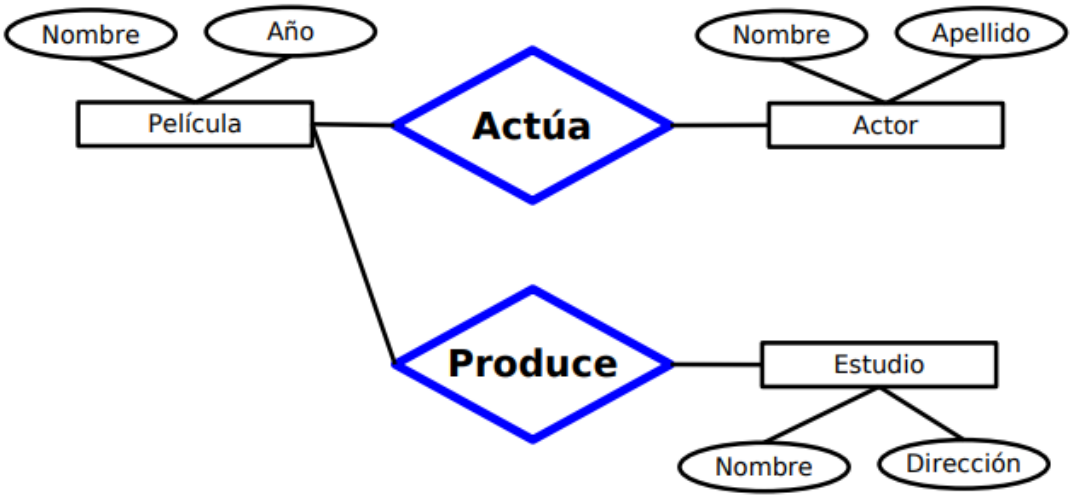
En este ejemplo, nombre, año, etc… son atributos de la entidad película.



Relación

Una relación es una asociación entre diferentes entidades.

Por ejemplo, la relación entre película y actor es que, el actor **actúa** en una película y un estudio **produce** una película.



Entonces se pudo observar que en los diagramas anteriores existen:

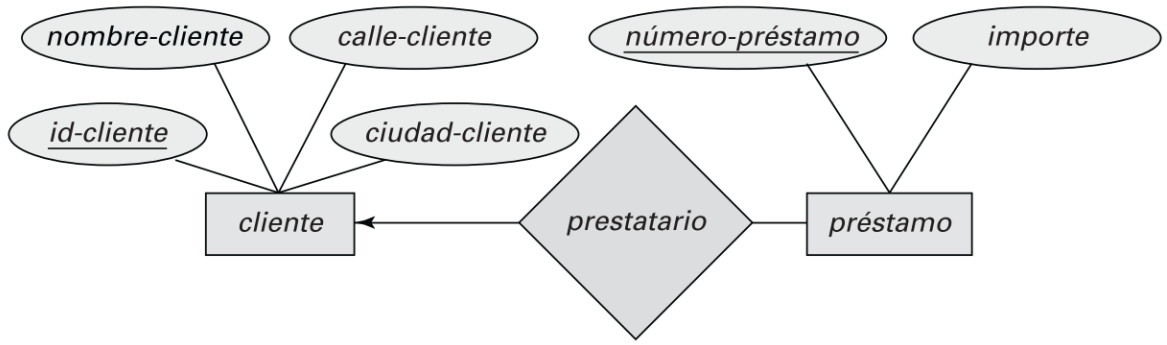
- Rectángulos, que representan conjuntos de entidades.
- Elipses, que representan atributos.

- Rombos, que representan relaciones.
- Líneas, que unen atributos y a entidades

Restricciones

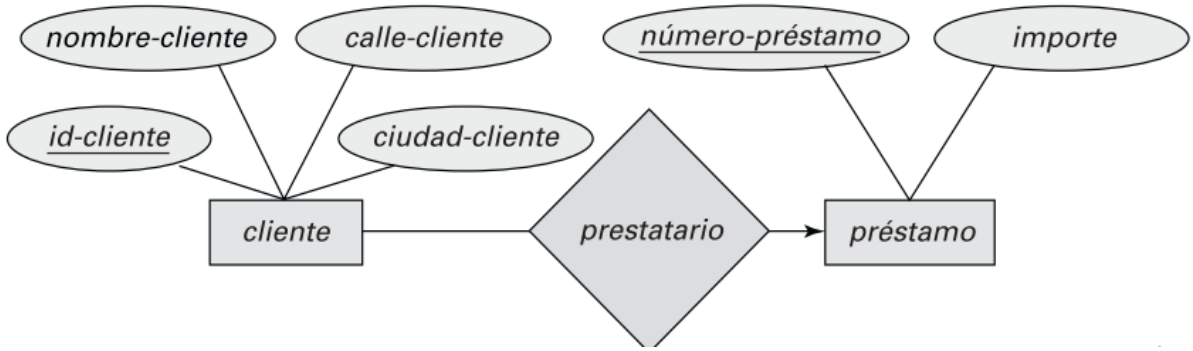
Son reglas que deben mantener los datos almacenados en la base de datos.

Relación de uno a muchos

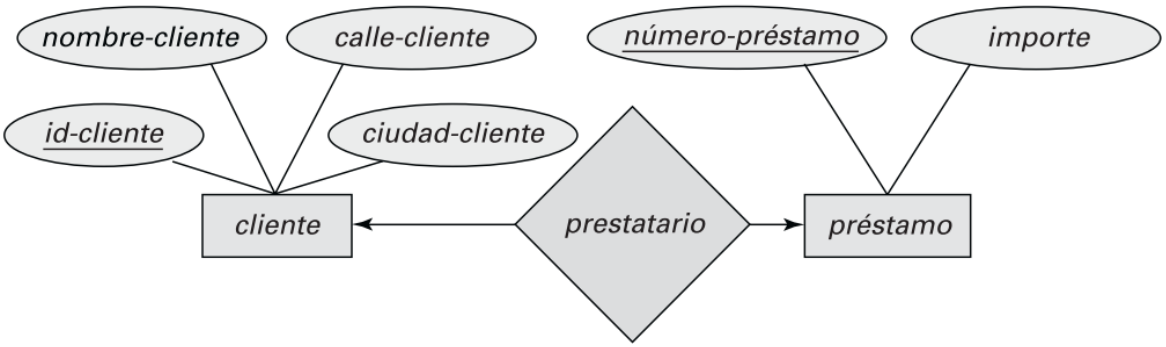


Un cliente tiene muchos préstamos

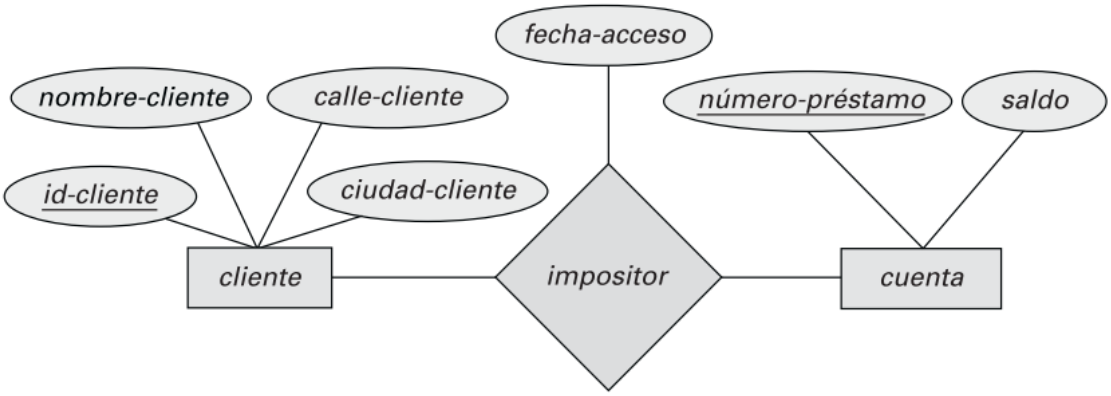
Relación de muchos a uno



Relación de muchos a muchos



También se puede agregar atributos a la relación prestataria que se convierte en entidad, porque la relación es de muchos a muchos.

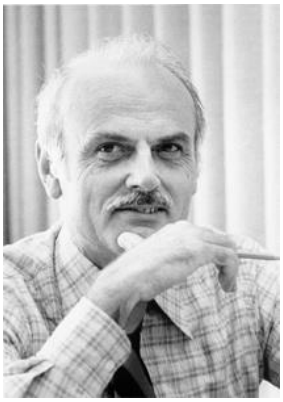


Resumen

Concepto	Descripción
Entidad	Las entidades representan cosas u objetos (ya sean reales o abstractos)
Atributo	Los atributos definen o identifican las características de entidad (es el contenido de esta entidad).
Relación	Una relación es una asociación entre diferentes entidades.

Restricción	
Relación de uno a uno	Una entidad se relaciona únicamente con otra y viceversa.
Relación de uno a muchos	Determina que un registro de una entidad puede estar relacionado con varios de otra entidad, pero en esta entidad existir solo una vez.
Relación de muchos a muchos	Determina que una entidad puede relacionarse con otra con varios registros y viceversa.

TEMA 5: MODELO RELACIONAL



EDGAR FRANK CODD

Científico informático Ingles

VER VIDEO

La teoría del modelo de datos relacional es obra del investigador de IBM Edgar Codd en 1970. Goza de una fuerte base matemática. El modelo relacional se caracteriza a muy grandes rasgos por disponer que toda la información debe estar contenida en tablas, y las relaciones entre datos deben ser representadas explícitamente en esos mismos datos.

Una base de datos relacional es una colección de relaciones que contienen los datos que describen un ambiente de negocios

Sus componentes son los siguientes:

Relación → Tabla

Codigo	Nombre	Edad	Genero	Telefono
259875	Carlos	18	m	3114123456
259040	Jorge	19	m	3126543211
256734	Ana	18	f	3001298765
250021	Maria	20	f	3109977564

Llave primaria

Una relación siempre tiene una llave primaria, una llave primaria (llamada clave) es un atributo o un grupo de atributos en los que los valores son únicos en todas las tuplas (filas) de la relación.

Estudiante

Codigo	Nombre	Edad	Genero	Telefono
259875	Carlos	18	m	3114123456
259040	Jorge	19	m	3126543211
256734	Ana	18	f	3001298765
250021	Maria	20	f	3109977564

Llaves candidatas

Cuando una relación tiene más de un atributo (o grupos de atributos) que representen las tuplas como únicas, se debe escoger una como llave primaria. Ejemplo:

Codigo	Nombre	Edad	Genero	Telefono	Cedula	PAPA
259875	Carlos	18	m	3114123456	121212	3.8
259040	Jorge	19	m	3126543211	343434	3.5
256734	Ana	18	f	3001298765	767756	4.1
250021	Maria	20	f	3109977564	367787	4.0

Llave primaria LLaves Candidatas

Llaves foráneas

Son llaves que son primarias en una relación, y aparecen como atributos en otra relación, entonces se dice que son llaves foráneas para la relación en la que aparecen como atributos.

LABORATORIO

Id_lab	Nombre	Ubicacion	Horario	Id_laboratorista
259875	Redes	209	7-5	134324
259040	Desarrollo	203	7-5	343442
256734	Linux-Alife	206	7-5	567673
250021	Programación	119	7-5	789894

Atributo que es llave foránea en la relación Laboratorio

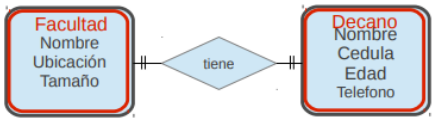
LABORATORISTA

Id_laboratorista	Nombre	Genero	Edad
259875	Juan	m	30
259040	Cesar	m	31
256734	Daniel	m	25
250021	Rosa	f	38

Relaciones

Relación de uno a uno

Son los atributos de una entidad que pasan a ser atributos de la otra tabla o relación. Una sola relación con los atributos de las dos entidades

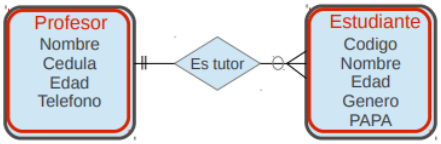


Facultad

Nombre	Ubicacion	Tamaño	Nombre_decano	Cedula_decano	Edad	telefono

Relación de uno a muchos

Atributo llave primaria de la entidad con cardinalidad uno pasa a ser llave foránea (atributo) de la entidad con cardinalidad múltiple. Se crean dos relaciones.



Estudiante

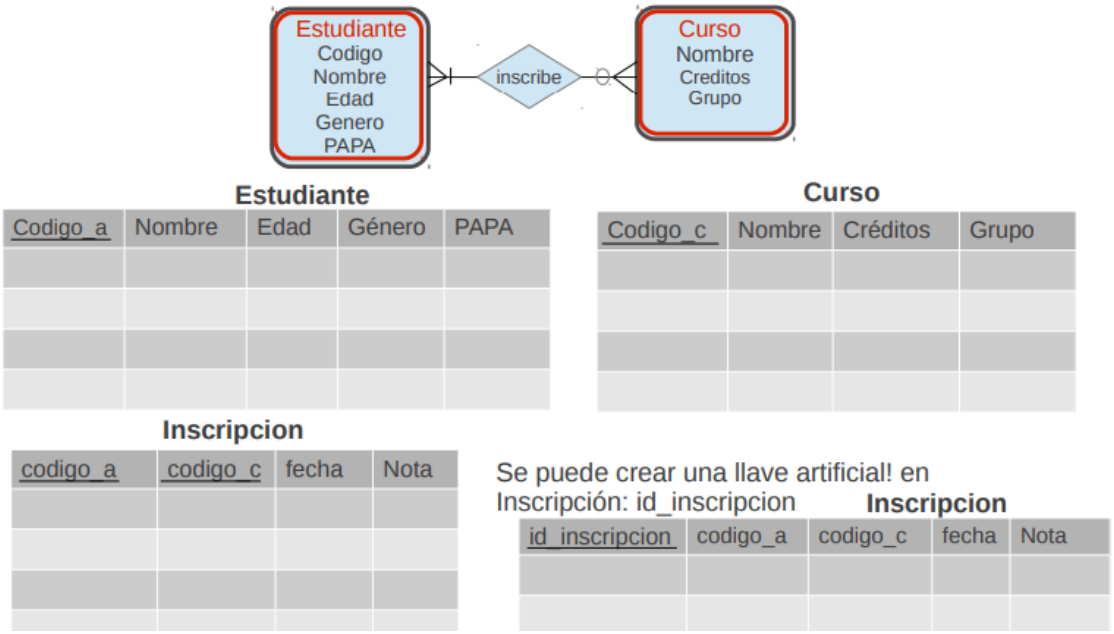
Codigo	Nombre	Edad	Género	PAPA	id_tutor

Profesor

cedula	Nombre	edad	telefono

Relación de muchos a muchos

La relación de muchos a muchos tiene como característica, el crear una nueva relación con el nombre de la relación de las dos entidades y con atributos las llaves primarias de las entidades relacionadas (llaves foráneas). La llave primaria de la nueva relación son el grupo de llaves foráneas, yo se crea una llave artificial.



En el ejemplo anterior solo se crea una de las tablas inscripción ya una `id_incripción` como clave artificial que dicho sea de paso lo recomiendo.

TEMA 6: NORMALIZACIÓN DE BASE DE DATOS

Introducción

Es importante comprender que una base de datos debe contar con alguna norma que verifique la calidad con la que fue diseñada y desarrollada, para ello esto en 1970 E. F. Codd, fue el creador de las bases de datos relacionales y a su vez acuño el término de normalización, según algunos autores:

Definición de normalización

- "La normalización es una técnica para diseñar la estructura lógica de los datos de un sistema de información en el modelo relacional"(Marqués, 2009)
- "Normalizar las tablas de una base de datos significa optimizar su diseño para no repetir datos innecesariamente y para prevenir problemas de inconsistencia." (Fuentes Gómez, 2013).

En síntesis, el proceso de normalización garantiza de cumplimiento de ciertas reglas en el diseño de una base de datos.

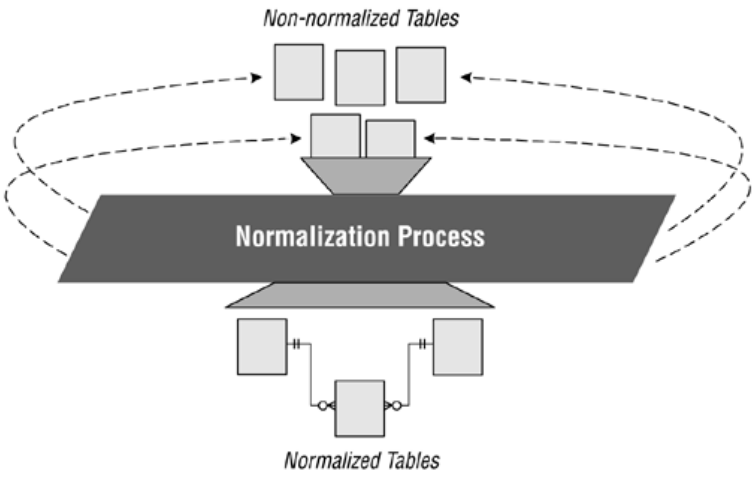


Figura: proceso de normalización

Como se muestra en el gráfico, las tablas relacionales ingresan a un proceso de normalización para luego cumplir con ciertos criterios de calidad y mejorar el rendimiento en su funcionamiento futuro.

Estas reglas son denominadas

1FN (primera Forma Normal).

2FN (segunda Forma Normal).

3FN (tercera Forma Normal).

La normalización tiene como ventajas principales:

La precisión. - Las interrelaciones entre las tablas consiguen mantener información diferente relacionada con toda exactitud.

La mínima redundancia. - la información no está duplicada innecesariamente.

Antes de considerar 3 formas normales se debe de definir el concepto de dependencia funcional para ello

Dependencia funcional

1FN. Primera Forma Normal

La 1FN es la forma mínima en la que se encuentran nuestras tablas. Consiste en añadir una tabla adicional para satisfacer los criterios mínimos de integridad, consistencia y concurrencia. Esta forma normal se utiliza para representar una relación básica y evitar grupos repetitivos. Una tabla está en 1FN si y solo si se cumplen las siguientes cinco condiciones:

Entonces, podríamos decir que la 1FN a controlar en las Bases de Datos las siguientes reglas:

- Eliminar grupos repetidos en tablas individuales.
- Crear una tabla diferente para cada conjunto de datos relacionados.
- Identificar cada conjunto de datos relacionados mediante una clave principal.
- No utilizar varios campos en una única tabla para almacenar datos similares.

Ejemplo

Supongamos que tenemos que guardar el nombre, apellidos y el teléfono de nuestros clientes. La primera idea sería crear una tabla así

id	nombre	apellidos	telefono
1	juan	calvimonte	33554815
2	Romeo	Julian Montes	33553575
3	Roxana	Cespedes Cardenas	33599815

Se adjunta el código sql

Pero entonces nos damos cuenta de que un cliente puede tener varios números de teléfono. Dos maneras poco eficientes de resolver esta necesidad serían las siguientes:

OPCIÓN 1: GUARDAR VARIOS REGISTROS EN EL MISMO CAMPO

id	nombre	apellidos	telefono
1	juan	calvimonte	33554815
2	Romeo	Julian Montes	33553575
3	Roxana	Cespedes Cardenas	33599815
4	Felipe	Melchor Castro	33889951 - 33445566
5	Jael	Alvarado Mamani	33326598 - 33445566 - 33554489
6	Alfredo	Colque Munca	33125689 - 33457816 - 33556928 - 33164976

Este diseño es probablemente la peor opción. Semánticamente no es claro, ya que no sigue la filosofía de la tabla. Además de complicar las futuras consultas a dicha tabla.

OPCIÓN 2: CREAR TANTAS COLUMNAS COMO CAMPOS NECESITEMOS

id	nombre	apellidos	telefono	telefono2
1	juan	calvimonte	33554815	(NULL)
2	Romeo	Julian Montes	33553575	(NULL)
3	Roxana	Cespedes Cardenas	33599815	(NULL)
7	Raquel	Cencini Rodriguez	33956230	33012587

Esta representación hace uso de columnas que permiten valores nulos, y por lo tanto no se conforman con la definición de la 1FN.

Por tanto, la manera correcta es adherir una nueva tabla a nuestra tabla clientes, relacionándose entre ellas mediante el ID único de cliente.

Así evitamos grupos repetidos y campos vacíos. Cada cliente podrá tener, o no, varios teléfonos.

2FN. Segunda Forma Normal

Una tabla que está en la primera forma normal (1NF) debe satisfacer criterios adicionales para calificar para la segunda forma normal. Específicamente: una tabla 1NF está en 2NF si y solo si, dada una clave primaria y cualquier atributo que no sea un constituyente de la clave primaria, el atributo no clave depende de toda la clave primaria en vez de solo una parte de ella. En otras palabras, lograríamos decir que la 2FN de Bases de Datos ha de controlar, en las Bases de Datos, las siguientes reglas:

- Crear tablas independientes para conjuntos de valores que se apliquen a varios registros.
- Relacionar dichas tablas mediante una clave externa.
- Los registros tan sólo deben depender de la clave principal de una tabla (si es necesario, puede ser una clave compuesta).

3FN. Tercera Forma Normal.

Una tabla está en 3NF si y solo la tabla está en la segunda forma normal (2NF), y ningún atributo no-primario de la tabla es dependiente transitivamente de una clave primaria. La mayoría de las tablas 3NF están libres anomalías de actualización, inserción, y borrado, pero ciertos tipos de tablas 3NF, que en la práctica raramente se encuentran, son afectadas por tales anomalías. En palabras más simples, en 3NF eliminaremos los atributos de cada una de las entidades que no cumplan con la segunda forma normal.

4FN. Cuarta Forma Normal.

La 4NF se asegura de que las dependencias multivaloradas independientes estén correcta yeficientemente representadas en un diseño de base de datos. La Redundancia es suprimida en la Cuarta Forma Normal. Una tabla está en 4NF si y

solo si está en Tercera forma normal o en BCNF (Otra Forma Normal de las Bases de Datos) y no posee dependencias multivaloradas no triviales. La definición de la 4NFconfía en la noción de una dependencia multivaloradas. En la 4FN, los campos eliminados en la Tercera forma normal se evalúan y si son necesarios los convertiremos en nuevas entidades, y aplicaremos nuevamente todas las formas normales.

5FN. Quinta Forma Normal.

La 5FN, es también conocida como Forma Normal de proyección-unión (PJ/NF). Esta Forma Normal esta designada para reducir redundancia en las bases de datos relacionales que guardan hechos multivalorados aislando semánticamente relaciones múltiples relacionadas. Entonces, podemos aseverar que la Quinta Forma Normal está caracterizada por: La tabla está en 5 FN si está en 4FN.No existen relaciones de dependencias no triviales que no siguen los criterios de las claves. Una tabla que se encuentra en la 4FN se dice que está en la 5FN si, y sólo si, cada relación de dependencia se encuentra definida por las claves candidatas.

Conclusión.

La Normalización de las Bases de Datos originalmente era un concepto para estudiantes de la Universidad Harvard, pero actualmente está abierto a cualquier persona que tenga un proyecto de base de datos abierto. Gracias a la Normalización de las Bases de Datos, podemos ganar gran rapidez, estabilidad y eficacia, sin contar con que se hace más fácil su estudio y análisis. Cabe destacar que la decisión de definir entidades es del diseñador de la base de datos y la responsabilidad de tratar de que se cumplan las formas normales es, también, del diseñador de la base de datos, y en él esta definir los atributos necesarios y entidades

TEMA 7: Sentencias avanzadas de manipulación de datos

Revisión de sentencias comunes DML

Consultas simples y multitablas

Consultas de agregado, subconsultas

Sentencias mixtas: Select, Update, Insert

.

TEMA 8: Manejo de objetos con Laravel

BIBLIOGRAFÍA

Camps Paré, R., Casillas Santillán, L. A., Costal Costa, D., Gibert Ginesta, M., Escofet, C. M., & Pérez Mora, O. (2005). *Bases de datos (Software libre)*. Universitat Oberta de Catalunya.

Fuentes Gómez, M. D. C. (2013). Notas del curso Bases de Datos. In *Universidad Autónoma Metropolitana Unidad Cuajimalpa*. UNIVERSIDAD AUTÓNOMA METROPOLITANA.

Gómez Ballester, E., Martínez Barco, P., Moreda Pozo, P., Suárez Cueto, A., Montoyo Guijarro, A., & Saquete Boro, E. (2010). Bases de Datos I. In *Pueblo y Educación. La Habana*. Universidad de Alicante.

Harford, T. (2020). *Herman Hollerith, el hombre que se hizo rico con los datos un siglo antes que Google*. https://www.bbc.com/mundo/noticias-51052936

Marqués, M. (2009). *Bases de Datos*. Departamento de Ingeniería y Ciencia de la Computación.

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2002). *Fundamentos de Base de Datos* (4ta., Vol. 44, Issue 8). McGraw-Hill.